PASC audiocompressie

Het door Philips ontwikkelde PASC-algoritme waarmee de omvang van digitale audio-data aanzienlijk wordt gereduceerd is de basis van alle moderne systemen voor de streaming van digitale audio, zoals het MP3 audio-formaat en het MPEG-1 video-formaat.

Auteur: Jos Verstraten, Landgraaf, Nederland
Email: josverstraten@live.nl
Publicatiedatum: 03-11-2020

Achtergrond informatie

Audio-CD te omvangrijk voor het prille internet
Toen internet populair werd begon men na te denken over manieren om muziek via dit nieuwe medium digitaal te versturen. Het was snel duidelijk dat dé manier om audio in die tijd digitaal te verwerken, het audio-CD formaat, absoluut ongeschikt was voor dit nieuwe medium. Audio-CD werkt met 16 bit brede samples en met 44.100 samples per seconde. Dat betekent dat er in één minuut muziek van audio-CD kwaliteit 44.100 ● 2 ● 2 ● 60 = 10.584.000 bytes zitten. Immers: de omvang van de data-stream is gelijk aan het aantal samples maal twee bytes per sample maal twee kanalen stereo maal 60 seconden. Als u dit via het prille internet wilde downloaden, stel met een toendertijd modern 28k8 modem, dan zou u dat ongeveer 45 minuten aan download tijd kosten. En dit voor slechts één minuut muziek op audio-CD kwaliteit.
Er moest dus een manier verzonnen worden om die grote hoeveelheid digitale gegevens flink te reduceren, zonder al te veel concessies te doen aan de geluidskwaliteit.

Een revolutie in elektronische geluidsregistratie!
Gedurende de gehele evolutie van de elektronische geluidsregistratie, vanaf de ontwikkeling van de eerste grammofoon tot de introductie van de audio-CD, heeft men er steeds naar gestreefd aan '*werkelijkheids weergave*' te doen. Het elektrische signaal moest een zo getrouw mogelijke representatie zijn van de geluidsgolven die ooit het membraan van de microfoon aan het trillen hadden gebracht.
Men spreekt in dit verband van '*perfecte representatie*'.
Maar die perfecte representatie kon niet bij audio-streaming via het prille internet, vanwege de beperkte data-stroom die mogelijk was. Dus heeft men bij de ontwikkeling van audio-streaming voor een geheel nieuwe en revolutionaire aanpak gekozen. Geen '*perfecte representatie*', maar '*perceptieve representatie*'. Het elektrische signaal is **NIET** een zo getrouw mogelijke weergave van de originele geluidstrillingen. Het is een signaal dat het menselijke gehoor de indruk (de perceptie) geeft te luisteren naar dat wat ooit door de microfoon werd opgenomen.

PASC: Precision Adaptive Sub-band Coding
Op hetzelfde moment was men bij Philips bezig een digitale opvolger van de succesvolle compact cassette te ontwikkelen. Dat zou DCC worden, de '***D**igital **C**ompact **C**assette*', die in 1992 door Philips en Matsushita werd geïntroduceerd. Ook bij dit systeem was het nodig een zeer grote reductie in de data-stroom te realiseren mét behoud van de geluidskwaliteit.
Philips heeft hiervoor het PASC-algoritme ontwikkeld, letterwoord van '***P**recision **A**daptive **S**ub-band **C**oding*'. In het Nederlands te vertalen door '*nauwkeurige aangepaste codering in sub-banden*'.

PASC kwam uiteraard niet uit de lucht vallen, maar borduurde verder op onderzoek naar geluidsmaskering van de TU Eindhoven, het Franse CCETT, het Duitse Institut für Rundfunktechnik en het Duitse Fraunhofer Institut.
PASC bracht de data-rate van audio-CD terug naar een kwart, van 1,4 Mbit/s naar 384 kbit/s. Latere verdere ontwikkeling van het algoritme heeft data-reducties tot meer dan een factor tien opgeleverd!
De grote data-reductie die door PASC wordt geïntroduceerd zit enerzijds in een perceptieve benadering van het geluid en anderzijds in een aantal onschuldige maar zeer ingenieuze technische grapjes.
Het DCC-systeem van Philips is volledig geflopt, de principes die in het PASC-algoritme worden gebruikt zijn echter wereldwijd geaccepteerd en toegepast in vrijwel alle data-reductie systemen voor audio. De principes van PASC worden bijvoorbeeld toegepast bij het welbekende video-formaat MPEG-1 en het nog veel bekendere audio-formaat MP3.
Goed geïnformeerde bronnen beweren dat Philips ongeveer 500 miljoen dollar heeft verdiend met zijn patenten op PASC.

De PASC-principes voor data-reductie

Inleiding

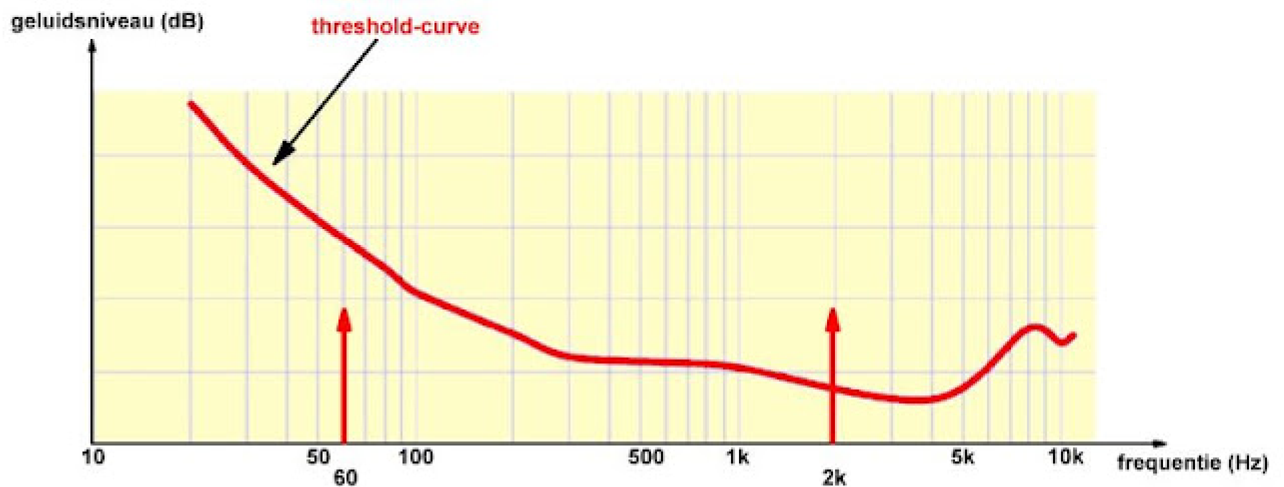
De data-reductie volgens het PASC-algoritme is gebaseerd op de onderstaande principes:

- De drempelwaarde van het menselijk gehoor.
- Invoeren van sub-banden.
- Het maskeereffect van het menselijke gehoor.
- Adaptive bit allocation.
- Middeling van het gemiddelde niveau.
- Re-allocatie van de codes.

Deze principes zullen in de volgende paragrafen worden besproken.

De drempelwaarde van het menselijk gehoor

Het is reeds lang bekend dat het menselijk gehoor frequentie selectief gevoelig is. Dat betekent dat er een gehoorcurve kan worden opgesteld, die natuurlijk niets meer wil zijn dan een gemiddelde van duizenden luisteraars. Deze curve is getekend in de onderstaande figuur. De getekende curve geeft het geluidsniveau in dB aan dat bij iedere frequentie nodig is om waargenomen te kunnen worden door de gemiddelde menselijke luisteraar. Deze lijn noemt men de '*threshold*', de lijn die aangeeft waar voor iedere frequentie de drempelwaarde van het oor ligt. Geluiden die zwakker zijn dan deze drempelwaarde worden niet waargenomen. Uit deze grafiek volgt dat ons oor het gevoeligst is voor frequenties tussen 2 kHz en 5 kHz. Naar beneden en naar boven neemt de gevoeligheid van het oor sterk af. Volgens de verzinners van het PASC-principe is het dan ook onzin om dergelijke geluiden, die toch niet worden gehoord, digitaal te coderen. Zou PASC een signaal moeten digitaliseren dat bestaat uit twee even grote sinussen met frequenties van 60 Hz en 2 kHz, zoals weergegeven in de grafiek, dan zou PASC het signaal van 60 Hz weglaten en niet digitaliseren. Het signaal van 60 Hz ligt immers onder de gehoorgrens.

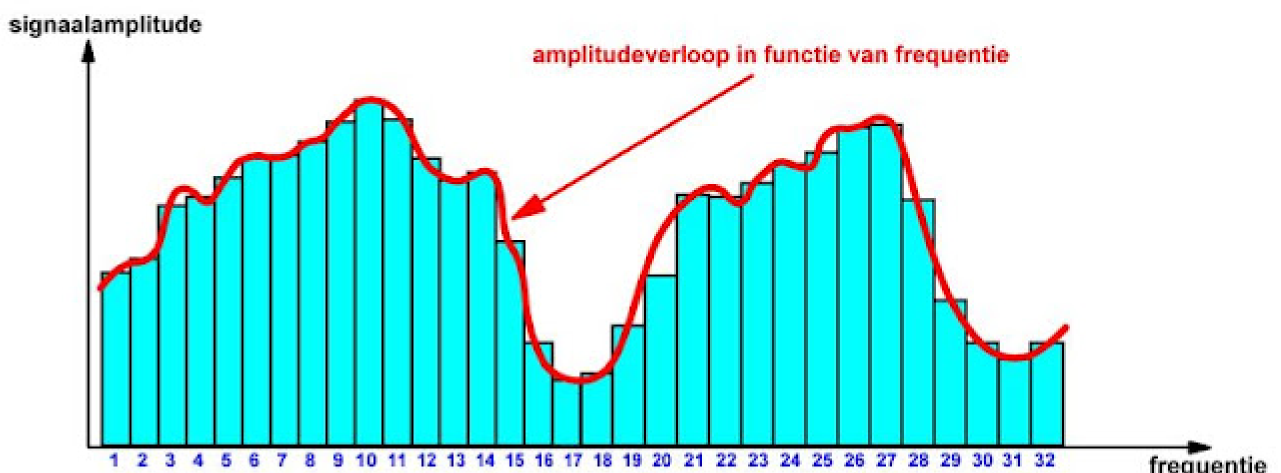


De gehoorcurve van het menselijk gehoor. (© 2020 Jos Verstraten)

Natuurlijk is dit principe niet zo eenvoudig in de praktijk te brengen. De drempelwaarde van het gehoor is immers geen statische waarde, maar werkt dynamisch. De drempel is namelijk afhankelijk van het gemiddelde geluidsniveau van het signaal. Vandaar dat PASC volledig dynamisch werkt en de drempelwaarde voor iedere frequentie steeds weer vergelijkt met het gemiddeld niveau van het signaal.

Invoeren van sub-banden van 600 Hz

In de praktijk is het uiteraard onmogelijk om dat vergelijkingsproces voor iedere frequentie uit de volledige frequentieband van 20 Hz tot 20 kHz uit te voeren. Vandaar dat het audio-bereik van 20 Hz tot 20 kHz in 32 sub-bandjes wordt ingedeeld, die ieder 600 Hz breed zijn. Deze bandjes zijn, zie de onderstaande figuur, lineair over de frequentie-as verdeeld.
Traditionele analoge elektronici zullen zo hun bezwaren hebben tegen dergelijke scherpe filtertechnieken. Vergeet niet dat de amplitude-inhoud van al deze bandjes nadien, bij het weergeven van het signaal, weer netjes tot één brede audio-band gecombineerd moeten worden! Met analoge schakelingen is dit volstrekt onmogelijk zonder grote problemen te krijgen met fase- en looptijdverschuivingen. Met moderne digitale signaalbewerkingstechnieken is dit echter zonder kwaliteitsverlies mogelijk.
Nadien berekent het PASC-algoritme voor iedere sub-band de verhouding tussen gemiddeld signaalniveau en een vast geprogrammeerde gehoordrempel voor deze band. Ligt de amplitude-inhoud van een sub-bandje onder de gehoordrempel, dan wordt de inhoud van dat bandje niet gecodeerd.

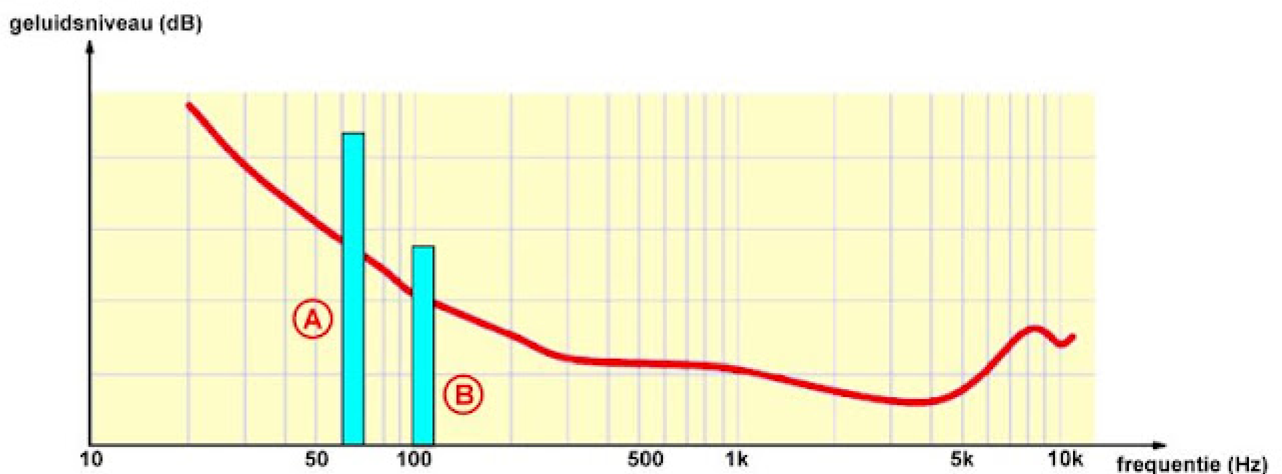


De 32 digitale sub-bandjes met een bandbreedte van 600 Hz. (© 2020 Jos Verstraten)

Het maskeereffect van het menselijke gehoor

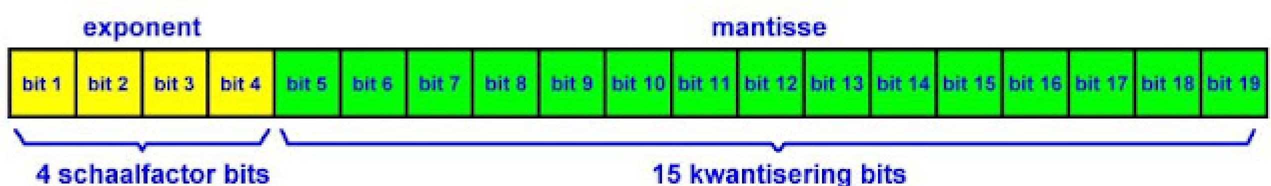
Uit zeer uitgebreide luisterproeven is gebleken dat het menselijke gehoor een bepaald maskeereffect heeft. Dit verschijnsel wordt toegelicht aan de hand van de grafiek in de onderstaande figuur. Een sub-band A met een grote inhoud kan een in frequentie vlakbij

gelegen zwakkere sub-band B maskeren. Dat wil zeggen dat het gehoor dat tweede signaal niet zal waarnemen, ondanks dat de amplitude ervan boven de gehoordrempel ligt. Het lijkt erop alsof het sterke signaal tijdelijk een plaatselijke vervorming van de threshold-curve veroorzaakt, waardoor alle frequenties in de omgeving van het sterke signaal opeens minder hoorbaar worden.
Een tweede maskeereffect ontstaat in de tijd. Als een zacht geluid een paar milliseconde eerder door een hard geluid is voorafgegaan zal het zachte geluid niet waarneembaar zijn. Het lijkt er op alsof het harde geluid ons gehoor eventjes verdooft.
Het PASC-algoritme houdt rekening met deze twee maskeereffecten. De software voert een continue analyse uit van het amplitudeverloop in ieder van de 32 sub-bandjes, vergelijkt de gegevens van een sub-band met de analyses van de aangrenzende sub-bandjes en analyseert bovendien het amplitudeverloop in functie van de tijd. Aan de hand van al deze gegevens besluit het PASC-algoritme welke signalen wel en welke signalen niet verder worden verwerkt.
Dank zij deze zeer intelligente werkwijze zorgt het systeem er toch voor dat de subtiele klank van een triangel niet wordt gemaskeerd door een paukslag. De frequenties van beide instrumenten liggen immers zo ver uit elkaar dat deze niet in aangrenzende sub-bandjes liggen en dus niet voldoen aan de PASC-criteria voor maskering.



Het maskeereffect van het menselijk gehoor. (© 2020 Jos Verstraten)

Adaptive bit allocation
De volgende stap is dat alles dat boven de gehoor- en maskeerdrempels ligt digitaal verwerkt wordt. Het audio-CD systeem gebruikt hiervoor een zogenoemde '*lineaire kwantisering*'. Ieder monster wordt standaard omgezet in een 16 bit brede digitale code. Ieder monster bevat dus even veel geluidsbits. PASC werkt met een zogenoemde '*floating point kwantisering*'. Voor ieder monster staan in principe 19 bits ter beschikking. In principe, omdat ook dit een dynamisch proces is en het PASC-algoritme alleen zoveel bits aan een monster toewijst als nodig voor het kwantiseren van het monster. Voor het ene monster kan dat noodzakelijke aantal bits inderdaad 19 bedragen, voor een volgend monster kan het echter best zijn dat 6 bits volstaan.
De 19 bits worden ingedeeld in twee groepen. Vier bits bepalen een zogenoemde '*schaalfactor*', de overige vijftien bits staan ter beschikking voor de kwantisering van het monster. Dit wordt toegelicht in de onderstaande figuur. De 4 bits voor de schaalfactor noemt men ook wel eens de '*exponent*' van de code, de 15 bits voor de kwantisering de '*mantisse*'.

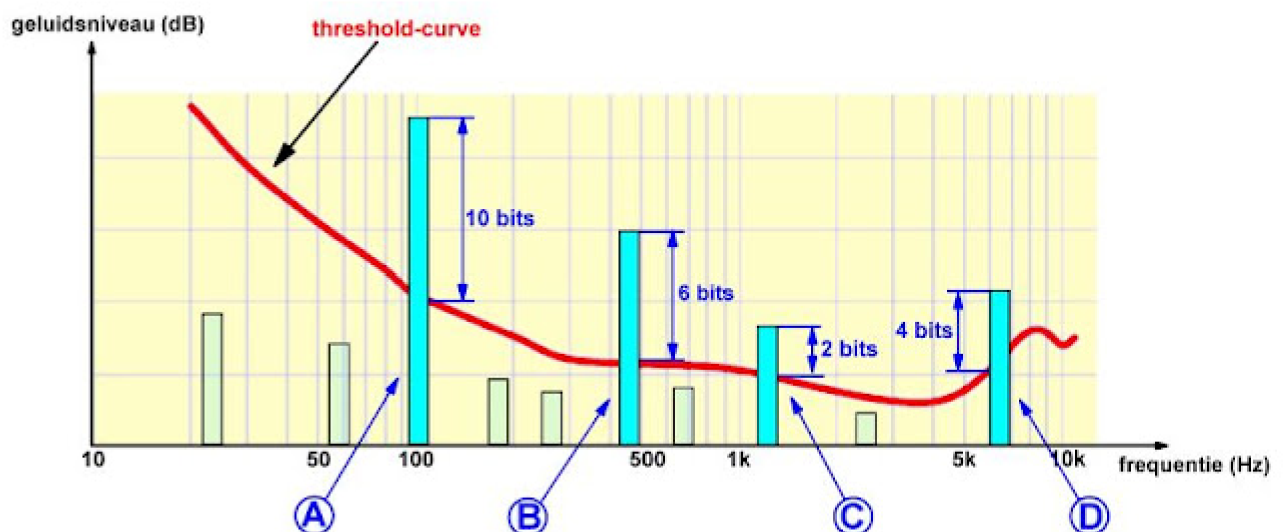


De bitstructuur van de PASC-codering van één monster. (© 2020 Jos Verstraten)

De signaal/ruis-afstand

Aan de hand van deze maximale ruimte voor het kwantiseren van een monster berekent Philips de maximale theoretische signaal/ruis-afstand van het PASC-systeem als 92 dB. Maar dank zij de later te bespreken dynamische manier waarop met deze 19 bits wordt omgesprongen zou men kunnen stellen dat de praktische signaal/ruis-afstand gelijk is aan 108 dB. Deze waarde ligt niet minder dan 10 dB boven de maximale 98 dB die het audio-CD systeem te bieden heeft!

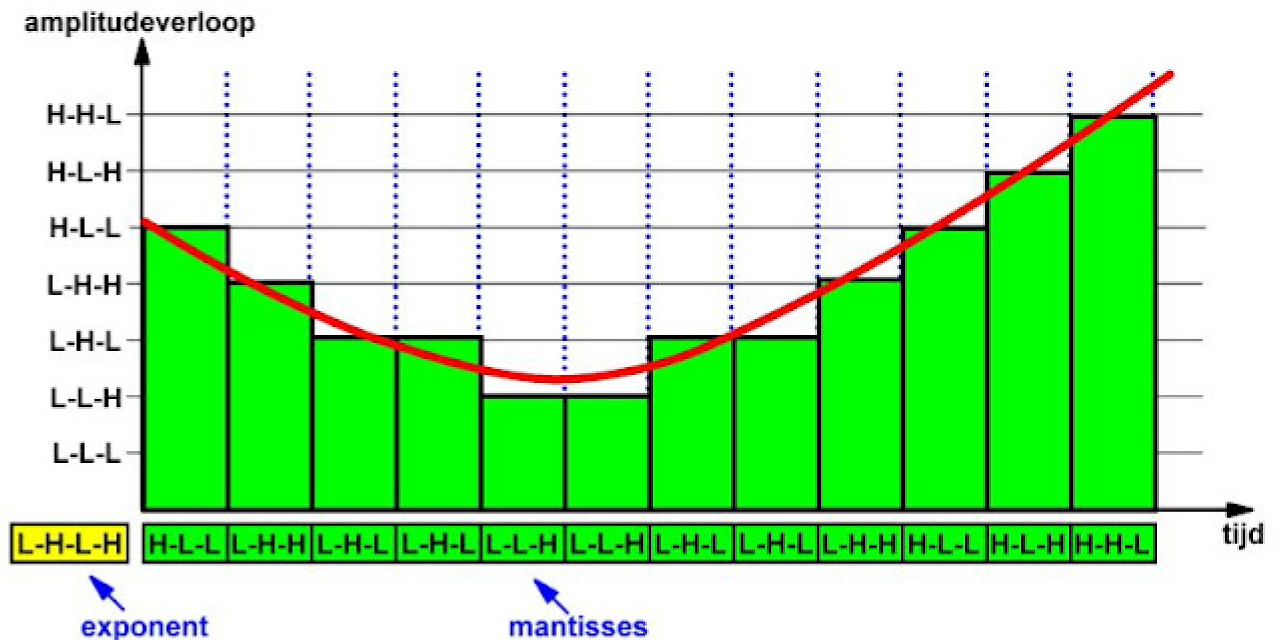
Dynamisch toekennen van de bits
Zoals gezegd worden de 4 + 15 bits dynamisch toegekend. Hoe dat gaat is toegelicht in de onderstaande figuur. In deze figuur worden tien monsters beschouwd, die in tien verschillende sub-bandjes liggen. De gestippeld getekende monsters worden niet gekwantiseerd omdat zij onder de drempels van het gehoor liggen. Blijven dus over de monsters A, B, C en D.
Het PASC-algoritme vindt het niet noodzakelijk om de totale amplitude van deze monsters te kwantiseren. Het gedeelte van de amplitude dat onder de threshold-curve ligt kan in één keer gedefinieerd worden door een schaalfactor in te voeren. Deze schaalfactor houdt dus rekening met de absolute amplitude van het signaal. De waarde van de schaalfactor wordt opgeslagen in de vier eerste bits. Nadien codeert PASC alleen dat deel van de amplitude dat boven de gehoordrempel ligt. Bovendien kijkt het systeem naar de grootte van deze amplitudes. Signaal C zit bijvoorbeeld maar iets boven de gehoordrempel en het zou zonde zijn om deze kleine amplitude in 15 bits te digitaliseren. Dat kan net zo goed met slechts 2 bits. De te kwantiseren amplitude van signaal A is echter groot en PASC stelt daarvoor dan ook 10 bits ter beschikking. Zo zal het dus maar zelden voorkomen dat PASC de toegestane 15 bits nodig heeft om een amplitude te kwantiseren, hetgeen een aanzienlijke code-reductie tot gevolg heeft.



De dynamische bit toekenning aan diverse monsters bij het PASC-systeem.
(© 2020 Jos Verstraten)

Middeling van het gemiddelde niveau
Een verdere data-reductie wordt bereikt door de digitale codes van twaalf opeenvolgende monsters in een zogenoemd '*PASC-raam*' samen te voegen. Onderzoeken hebben uitgewezen dat de gemiddelde waarde van een geluidssignaal, vergeleken met de samplingfrequentie, maar zeer langzaam varieert. Dat wil zeggen dat de opeenvolgende samples meestal dezelfde waarde voor de schaalfactor hebben. Het is dus niet noodzakelijk deze waarde in de code van ieder monster te verwerken. Vandaar dat PASC de gemiddelde signaalwaarde over een periode van 12 samples berekent en deze eenmalig in een digitale schaalfactor onderbrengt. Nadien volstaat het voor deze 12 monsters alleen de waarden van de kwantisaties in de code op te nemen. Dit principe wordt in de onderstaande figuur toegelicht aan de hand van een voorbeeld. Het analoge signaal schommelt in dit voorbeeld rond een bepaalde gemiddelde waarde. PASC berekent deze waarde en stelt de schaalfactor voor deze 12 monsters vast op 'L-H-L-H'.

Nadien berekent PASC de amplitude-afwijking van ieder monster ten opzichte van de gemiddelde waarde.
Duidelijk blijkt dat de momentele waarde erg weinig varieert.
Er zijn steeds slechts 3 bits noodzakelijk om het amplitudeverloop ten opzichte van de schaalfactor te definiëren. Voor het volledig definiëren van deze 12 monsters heeft PASC dus slechts
4 + (12 ● 3) = 44
bits nodig, heel wat minder dan de
2 ● 16 = 192
bits die een systeem met lineaire kwantisering er voor nodig heeft.



Middeling van het gemiddelde niveau. (© 2020 Jos Verstraten)

Re-allocatie van de codes
Een van de eigenschappen van geluid is dat de frequentiesamenstelling voortdurend varieert. Het zal niet vaak voorkomen dat er een monster genomen wordt waarbij alle 32 sub-bandjes gevuld zijn. Integendeel, bij de meeste monsters zal het gebeuren dat diverse sub-bandjes geen informatie boven de gehoor- en maskeerdrempels bevatten.
Het zou nu niet efficiënt zijn om in de uiteindelijke digitale code van het monster ruimte te reserveren voor het opslaan van subcodes van sub-bandjes die toch leeg zijn. Vandaar dat het PASC-algoritme alweer zeer dynamisch de uiteindelijke code toewijst. Dit wordt '*re-allocatie*' genoemd. PASC verdeelt lange kwantiseringscodes ook over de lege sub-bandjes, waardoor de uiteindelijke code veel minder ruimte in beslag neemt.
Natuurlijk is het dat wel noodzakelijk de kwantiseringscode aan te vullen met een vorm van adrescodering. Anders zou de PASC-processor in de ontvanger bij het herwinnen van het analoge signaal uiteraard niet weten waar alle gegevens van een monster zitten.

Conclusie

Gemiddeld heeft het PASC-algoritme slechts vier bits nodig voor het digitaliseren van een sub-band monster. Toch wordt hiermee dezelfde resolutie bereikt als met een traditionele lineaire kwantisering die uit 16 bits bestaat! Omdat uiteraard met stereo gewerkt wordt bestaat ieder monster gemiddeld uit 8 bits. Dit betekent dus een data-reductie van 32 naar 8 bits.

Van PASC naar MP3

Inleiding
Het MP3-bestandsformaat is ontworpen door een groep onderzoekers rond de Duitser Karlheinz Brandenburg, onder leiding van diens promotor, en andere onderzoeksgroepen. In 2000 ontving hij hiervoor de Duitse Toekomstprijs. Het MP3-algoritme borduurt verder op de PASC-principes en maakt ook gebruik van de indeling van het audio-spectrum in 32 sub-bandjes en van perceptieve representatie. Om nog grotere data-reductie te verkrijgen heeft men een aantal extra technieken ingevoerd.

Selecteerbare bit-rate
Bij het MP3-formaat is het mogelijk om zélf de mate van compressie en dus van geluidskwaliteit te bepalen. Een belangrijke factor hierbij is de bit-rate, het aantal bits per seconde dat nodig is om de audio te verwerken. Veel gebruikte bit-rates zijn 64, 128, 192, 256 en 320 kilobit per seconde (kb/s). Hoe hoger de bit-rate, hoe hoger de kwaliteit en hoe minder compressie er is.
Zoals in het begin van dit artikel werd berekend heeft de lineaire digitalisering van het audio-CD formaat een bit-rate van 1.411.200 bit per seconde. Een MP3-bestand heeft daarentegen slechts 128.000 bit per seconde nodig om muziek met een voor het grote publiek aanvaardbare kwaliteit te digitaliseren. Dat levert een compressie-factor op van meer dan elf!

Constant Bit Rate versus Variable Bit Rate
PASC is een systeem dat werkt met een '***C**onstant **B**it **R**ate*', ook CBR genoemd. De volledige audio-opname wordt verwerkt met een vaste bit-rate. Bij MP3 heeft men echter het begrip '***V**ariable **B**it **R**ate*' (VBR) ingevoerd. Bij VBR wordt de bit-rate bepaald door de complexiteit van het audio-signaal. Stel dat er één seconde stilte in de muziek voorkomt. Die stilte kan worden gedigitaliseerd met een heel lage bit-rate, bijvoorbeeld 16 kb/s zonder dat er kwaliteitsverlies optreedt. De volgend seconde muziek kan dan echter weer heel ingewikkeld van samenstelling zijn en dan schakelt het MP3-algorimtme automatisch over naar een veel hogere bit-rate, bijvoorbeeld 192 kb/s.

Stoeien met de stereo-informatie
Er zit veel identieke informatie in het geluid van rechter en het geluid van het linker kanaal. Het MP3-algoritme zal het signaal maar één keer digitaliseren als het geluid van beide kanalen hetzelfde is.

De discrete cosinus transformatie (DCT)
Dit is een uiterst ingewikkelde wiskundige techniek die een grote data-reductie tot stand brengt. Het principe is een beetje te vergelijken met hoe een periodiek signaal volledig kan worden beschreven door de amplitudes en de frequenties van de diverse Fourier-frequenties waaruit het signaal bestaat te bepalen.

Entropie codering volgens Huffmann
MP3 maakt voor een extra data-reductie gebruik van de bekende '*entropie codering*' volgens het Huffmann algoritme. Huffmann was een Duitse wiskundestudent die als afstudeerproject de opdracht kreeg het meest efficiënte algoritme te ontwerpen om binaire gegevens te coderen. Dat deze man daarin meer dan geslaagd is mag blijken uit het feit dat het begrip '*Huffmann codering*' een standaard is geworden in de data communicatie. Bij de Huffmann codering wordt een seriële data-stroom onderzocht op het statistisch voorkomen van bepaalde codes. Codes die het vaakst aanwezig zijn worden nadien gecodeerd met slechts één bit. Codes die iets minder vaak aanwezig zijn worden gecodeerd met twee bits. De code die het minst aanwezig is wordt gecodeerd met de langste binaire code die het algoritme toelaat. In het reeds genoemde artikel 'Know-how: MPEG video compressie' wordt dit algoritme toegelicht aan de hand van een voorbeeld.

De MP3-tag
De MP3-tag is extra informatie die in de audio data-stream wordt verwerkt. In deze MP3-tag kan extra informatie over de muziek worden opgeslagen, zoals de naam van de artiest en de titel van het nummer. Het is zelfs mogelijk een lage-resolutie afbeelding van de artiest of van

een CD-hoes in deze tag digitaal te coderen.